Recursion

Objective

The objective of this lab is to understand the concept of Recursion. Another way to achieve repetition is through ***recursion***, which occurs when a function calls itself. To develop understandings implement the following recursive programs. Also manually trace each program to get better understanding of it.

Tasks

1. **Linear Recursive function**

Find factorial of a number?

**Explanation**

The factorial of a positive integer *n*, denoted *n*!, is defined as the product of the integers from 1 to *n*. If *n* = 0, then n! is defined as 1 by convention. More formally, for any integer *n* ≥ 0,



In general, for a positive integer *n*, we can define factorial(*n*) to be *n*.factorial(*n* − 1). This

leads to the following ***recursive definition***



1. **Recursive function**
2. Find the recursive function of the following code.

for i = 1 to 3

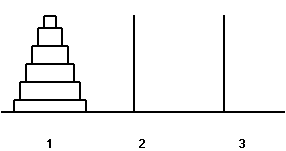
for j = 1 to 3

for k = 1 to 3

print i,j,k

1. Write a recursive function that has one parameter which is a size\_t value called x. The function prints x asterisks, followed by x exclamation points. Do NOT use any loops. Do NOT use any variables other than x.
2. **Binary Recursive function**

Solve the problem of tower of Hanoi as discussed in class.



You have three needles, need to move all disc from needle 1 to needle 2 using needle 3 as auxiliary. Constraints are as follows:

* Move one disc at a time.
* Always place smaller disc on larger one but not vice versa.

Algorithm will work as:

1. If (n==1) then move needle 1 to 2
2. If(n>1) then
   1. Move (n-1) disks from needle 1 to needle 3 using needle 2 as an auxiliary, so that largest disc could be accessed.
   2. Move this largest disc from needle 1 to 2.
   3. Now, Move (n-1) discs from needle 3 to 2 using needle 1 as an auxiliary.